AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) In a system that includes a user computer that communicates with a server computer over a network, a method for mitigating a cross-site scripting attack, the method comprising:

maintaining a list of active markers at a server;

receiving a request from a user computer, wherein the request includes a first portion of ~~safe~~ data, and a second portion of data, all of the second portion of data being derived from an outside source;

determining ~~if the~~ whether the request from the user computer includes a marker of active content identified in the list of active markers, wherein determining whether the request from the user computer includes a marker of active content includes examining only the second portion of data of the request;

refraining from ~~serving a response to~~ executing any portion of the request ~~if the~~when the request includes the marker of active content;

informing the user computer that a marker of active content from the list of active markers on the server has been discovered in the request; and

requesting that the user computer resubmit a request and subsequently serving a response to the request resubmitted by the user computer ~~if the~~ when the resubmitted request is unaccompanied by an marker of active content identified in the list of active markers.

2. (Original) A method as defined in claim 1, wherein receiving a request from a user computer further comprises receiving an HTTP request from the user computer.

3. (Original) A method as defined in claim 1, wherein receiving a request from a user computer further comprises at least one of:

receiving a cookie from the user computer;

receiving a query string from the user computer;

receiving an HTTP form from the user computer; and

receiving one or more HTTP headers from the user computer.

4. (Previously Presented) A method as defined in claim 3, wherein determining if the request from the user computer includes a marker of active content further comprises evaluating only the second portion of the request that includes the data derived from an outside source.

5. (Original) A method as defined in claim 1, wherein determining if the request from the user computer includes a marker of active content further comprises at least one of:

searching the request for one or more character combinations that correspond to a script construct;

searching the request for an event that includes a script construct; and

searching the request for an expression that includes a script construct.

6. (Original) A method as defined in claim 1, wherein determining if the request from the user computer includes a marker of active content further comprises searching the request for a pattern that indicates an unauthorized script.

7. (Previously Presented) A method as defined in claim 1, further comprising:

generating an event that is logged at the server; and

encoding a response that is delivered to the user computer informing the user computer of discovery of the marker of active content.

8. (Currently Amended) In a system that includes a user computer that communicates with a server computer over a network, wherein the server computer generates dynamic content based on input from the user computer, a method for mitigating a cross-site scripting attack such that data submitted to the server computer is not sent back to the user computer as script, the method comprising:

receiving an HTTP request at a server computer, wherein the HTTP request includes a ~~safe~~ first portion, and a user input portion, ~~that includes~~ the user input portion including user input data that was not generated by the server computer;

evaluating the HTTP request to determine ~~if the~~ whether the input data of the user input portion includes a script construct, wherein the script construct indicates that the HTTP request is part of a cross-site scripting attack, and wherein evaluating the HTTP request includes examining only the user input data;

refusing to ~~dynamically render a response to~~execute any portion of the HTTP request, thereby preventing the cross-site scripting attack ~~if the~~when the input data includes a script construct;

generating a notice indicating that a script construct indicative of a cross-site scripting attack has been received; and

requesting that a user resubmit an HTTP request, and subsequently executing and dynamically rendering a response to the HTTP request resubmitted by the user ~~if the~~ when the resubmitted HTTP request is free from script constructs in a user input portion of the resubmitted HTTP request.

9. (Original) A method as defined in claim 8, wherein receiving an HTTP request at a server computer further comprises at least one of:

receiving a query string that includes at least one query string variable;

receiving a cookie;

receiving one or more headers in the HTTP request; and

receiving one or more form fields.

10. (Original) A method as defined in claim 8, wherein evaluating the HTTP request to determine if the input data includes a script construct further comprises at least one of:

searching the HTTP request for one or more character combinations that correspond to a script construct;

searching the HTTP request for an event that includes a script construct;

searching server variables that derive input data from another source; and

searching the HTTP request for an expression that includes a script construct.

11. (Original) A method as defined in claim 8, wherein evaluating the HTTP request to determine if the input data includes a script construct further comprises searching the input data for a script construct.

12. (Original) A method as defined in claim 11, wherein searching the input data for a script construct further comprises searching for patterns associated with scripts.

13. (Cancelled).

14. (Original) A method as defined in claim 8, wherein preventing the cross-site scripting attack if the input data includes a script construct further comprises logging an event at the server computer.

15. (Original) A method as defined in claim 8, wherein preventing the cross-site scripting attack if the input data includes a script construct further comprises encoding the user input including the script construct to render the script inert.

16. (Original) A method as defined in claim 8, wherein evaluating the HTTP request to determine if the input data includes a script construct further comprises evaluating the HTTP request to determine in the input data includes a marker of active content.

17. (Original) A method as defined in claim 16, wherein evaluating the HTTP request to determine in the input data includes a marker of active content further comprises determining if the marker of active content is within a particular element, wherein the marker of active content is harmful only when rendered within the particular element.

18. (Currently Amended) In a system that includes a user computer that communicates with a server computer over a network, wherein the server computer generates dynamic content based on input from the user computer, a computer program product for implementing a method for mitigating a cross-site scripting attack such that input data submitted to the server computer is not sent back to the user computer as script, the computer program product comprising:

a computer-readable medium having computer executable instructions for performing the method, the method comprising:

receiving an HTTP request at a server computer, wherein the HTTP request includes a ~~safe~~ first portion and a user input portion that includes all user input data that was not generated by the server computer;

before performing dynamic rendering of a response to the HTTP request, evaluating the HTTP request to determine ~~if the~~ whether the input data of the user input portion includes a script construct that indicates a cross-site scripting attack, wherein evaluating the HTTP request includes examining only the user input portion of the HTTP request;

refusing to ~~dynamically render a response to~~execute any portion of the HTTP request, thereby preventing the cross-site scripting attack ~~if the~~ when the input data includes a script construct;

generating a notice indicating that a script construct indicative of a cross-site scripting attack has been received; and

requesting that a user resubmit an HTTP request, and subsequently executing and dynamically rendering a response to the HTTP request resubmitted by the user ~~if the~~when the resubmitted HTTP request is free from script constructs in a user input portion of the resubmitted HTTP request.

19. (Original) A computer program product as defined in claim 18, wherein receiving an HTTP request at a server computer further comprises at least one of:

receiving a query string that includes query string variables;

receiving a cookie;

receiving one or more headers in the HTTP request; and

receiving one or more form fields.

20. (Original) A computer program product as defined in claim 18, wherein evaluating the HTTP request to determine if the input data includes a script construct further comprises at least one of:

searching the HTTP request for one or more character combinations that correspond to a script construct;

searching the HTTP request for an event that includes a script construct;

searching server variables that derive input data from another source; and

searching the HTTP request for an expression that includes a script construct.

21. (Original) A computer program product as defined in claim 18, wherein evaluating the HTTP request to determine if the input data includes a script construct further comprises searching the input data for a script construct.

22. (Original) A computer program product as defined in claim 21, wherein searching the input data for a script construct further comprises searching for patterns associated with scripts.

23. (Cancelled).

24. (Original) A computer program product as defined in claim 18, wherein preventing the cross-site scripting attack if the input data includes a script construct further comprises logging an event at the server computer.

25. (Original) A computer program product as defined in claim 18, wherein preventing the cross-site scripting attack if the input data includes a script construct further comprises encoding the user input including the script construct to render the script inert.

26. (Previously Presented) A method as recited in claim 1, wherein determining if the request from the user computer includes a marker of active content comprises evaluating only user input fields of the request.

27. (Currently Amended) A method as recited in claim 1, wherein determining if the request from the user computer includes a marker of active content includes:

inactivating markers in the list of active markers.

28. (Previously Presented) A method as recited in claim 8, wherein evaluating the HTTP request to determine if the input data includes a script construct comprises evaluating the HTTP request for an onclick event.

29. (Previously Presented) A method as recited in claim 8, wherein evaluating the HTTP request to determine if the input data includes a script construct comprises evaluating the HTTP request for an element size